Football Match Result Prediction

Machine Learning Project | Bar-Ilan University

1. Introduction

This project explores football match prediction using machine learning techniques. The dataset is from Kaggle: https://www.kaggle.com/datasets/technika148/football-database. It includes top European league match statistics, with the goal of predicting match outcomes as Home Win, Draw, or Away Win.

2. Objective

The objective is to develop a machine learning model that can predict the outcome of a football match based on pre-game statistics and derived features.

3. Dataset

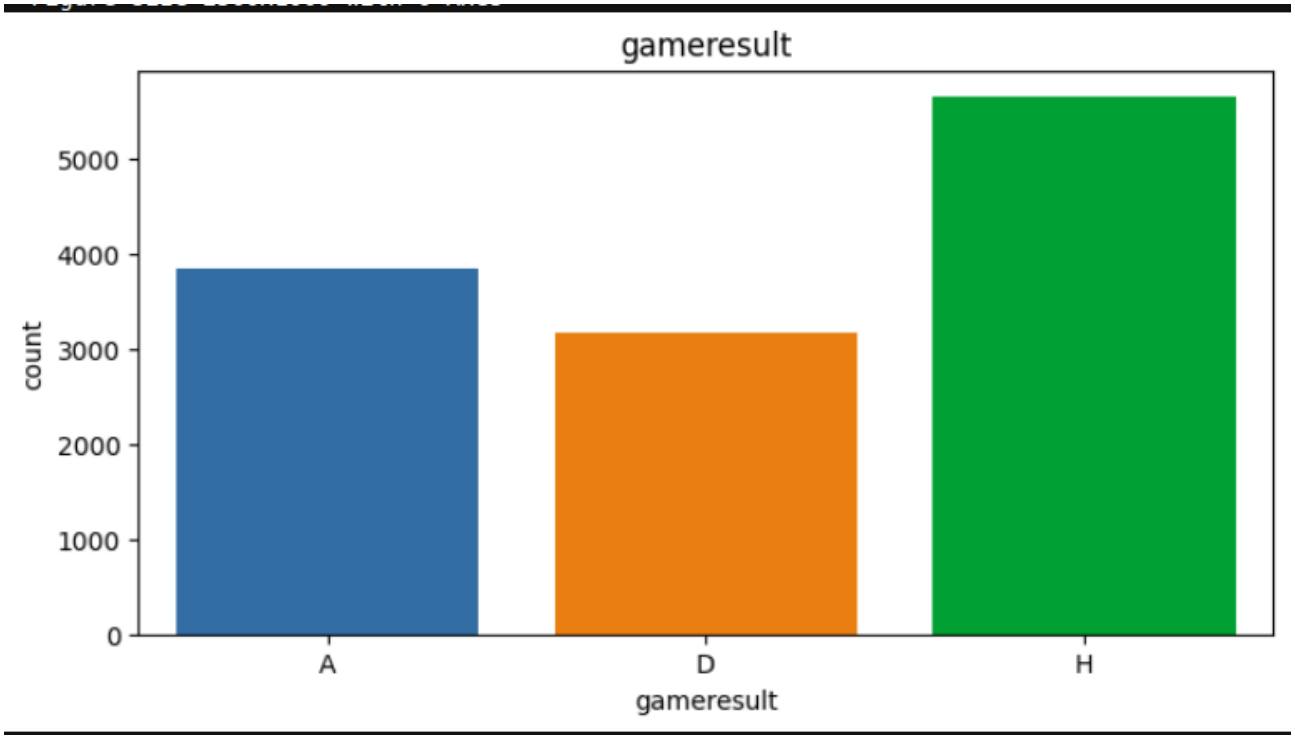
The dataset includes detailed match stats from the top 5 European leagues. Each row represents a match with over 100 features. Data includes goals, assists, cards, xG, fouls, and more. It was downloaded in CSV format from Kaggle.

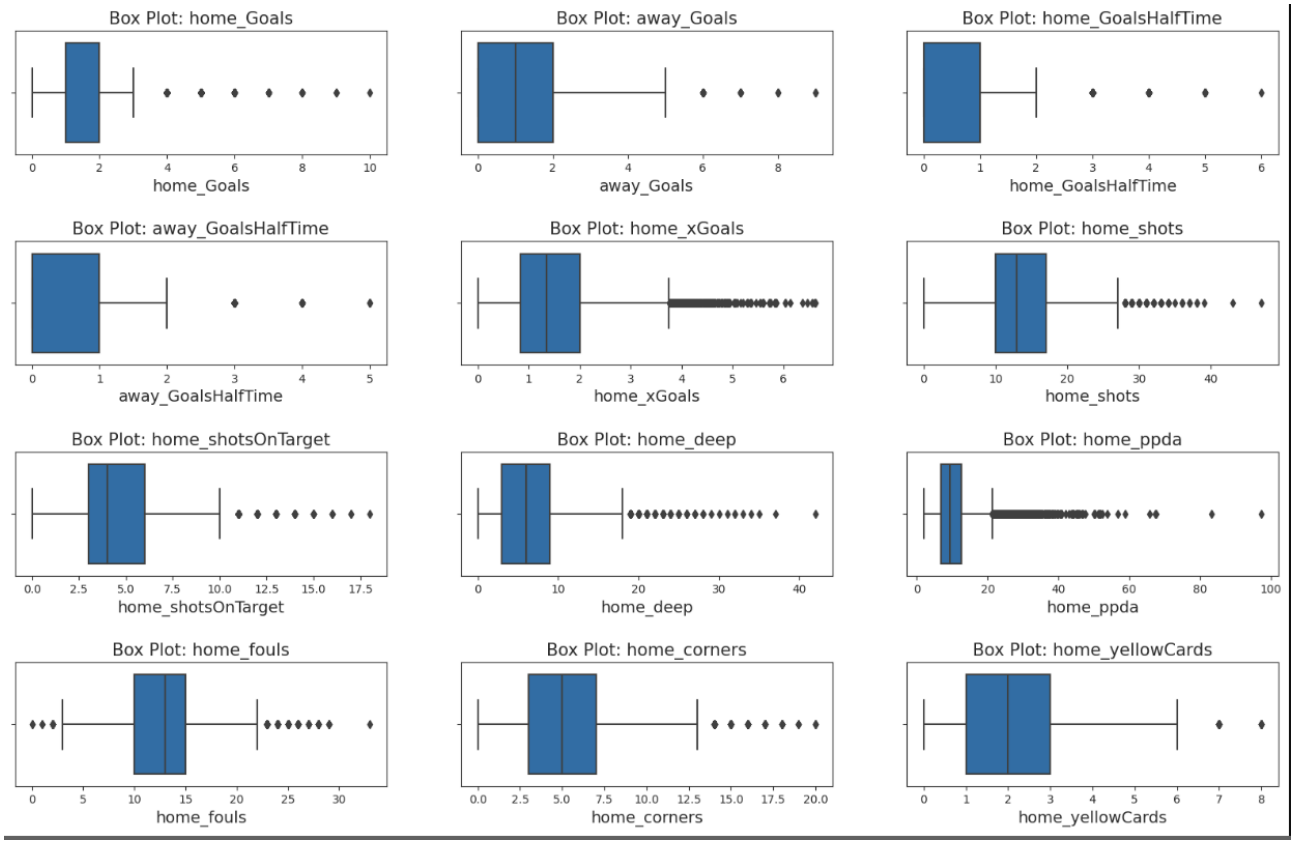
4. Project Design & Methodology

The project followed a structured pipeline: data preprocessing, EDA, handling outliers/missing data, feature engineering, feature selection, model training, hyperparameter tuning, and final evaluation.

Football Match Result Prediction

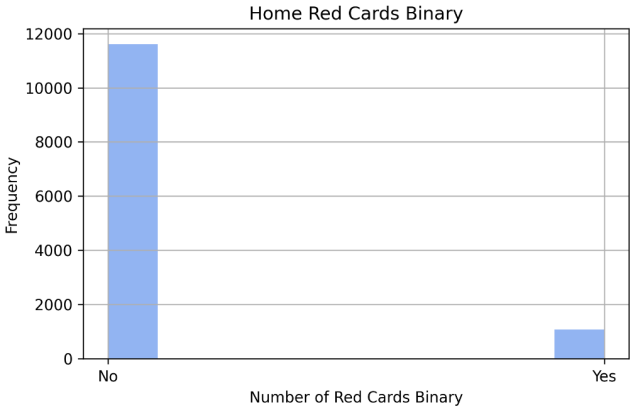
Machine Learning Project | Bar-Ilan University

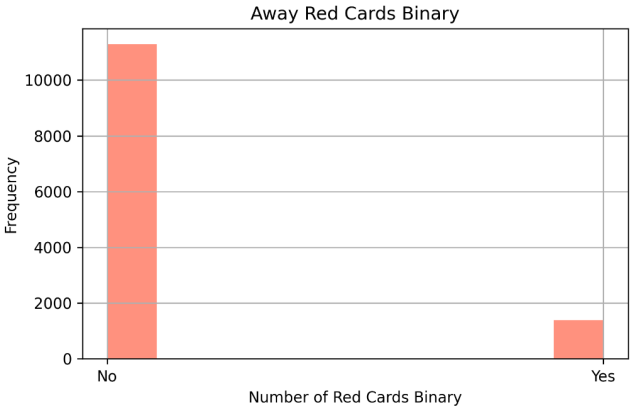


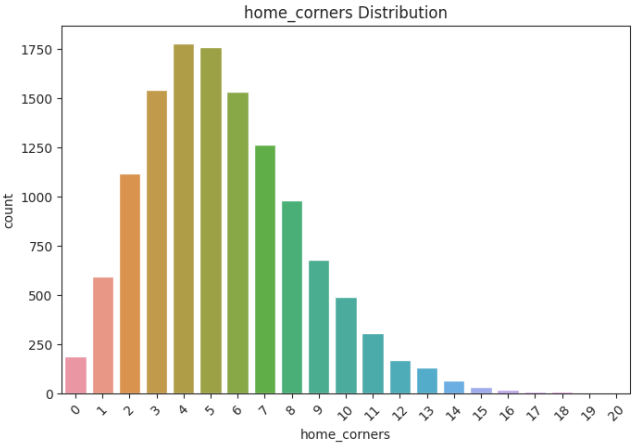


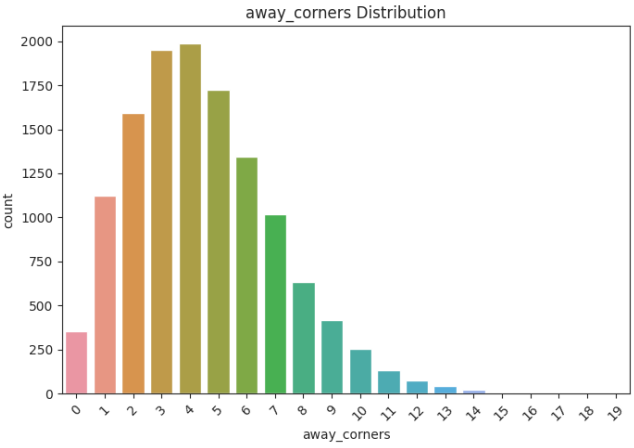
Football Match Result Prediction

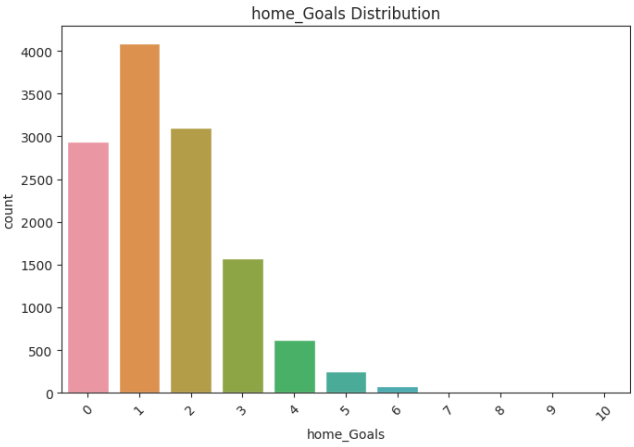
Machine Learning Project | Bar-Ilan University

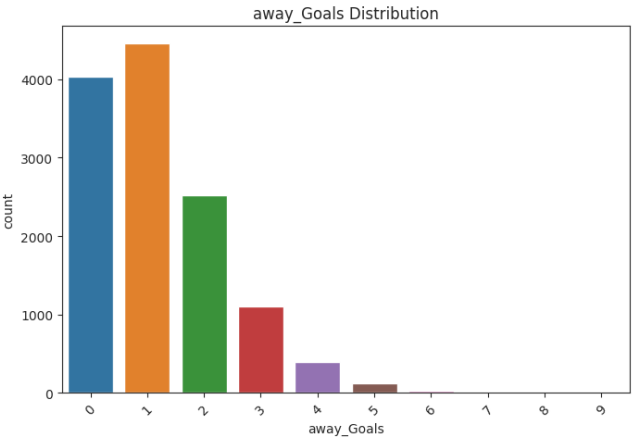






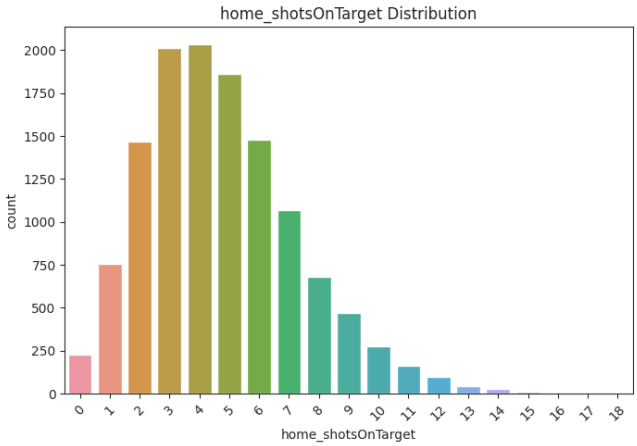


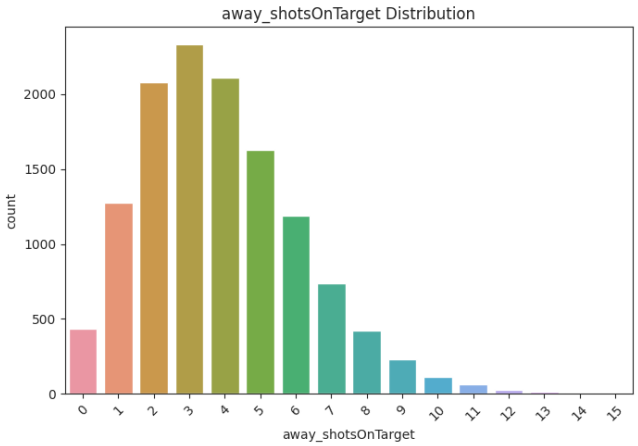


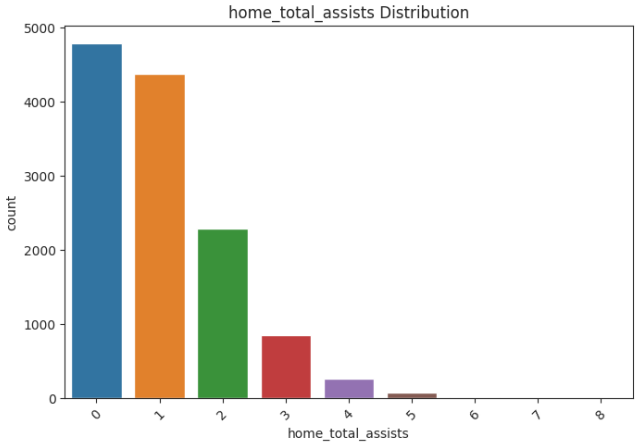


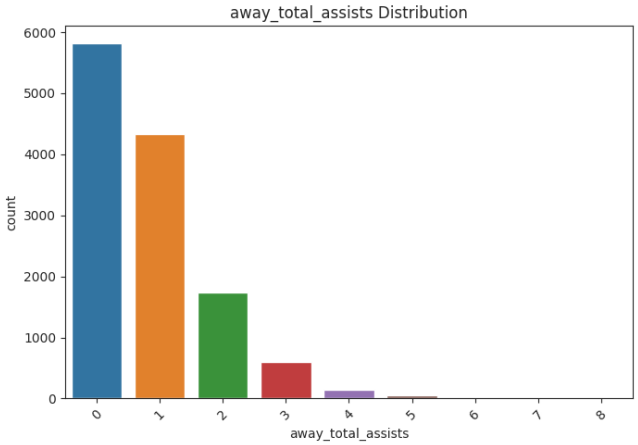
Football Match Result Prediction

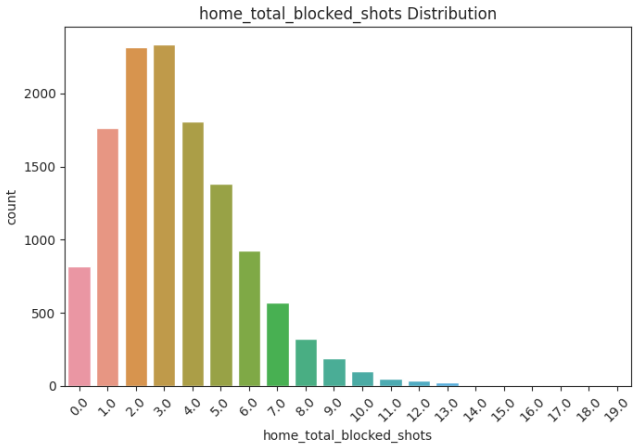
Machine Learning Project | Bar-Ilan University

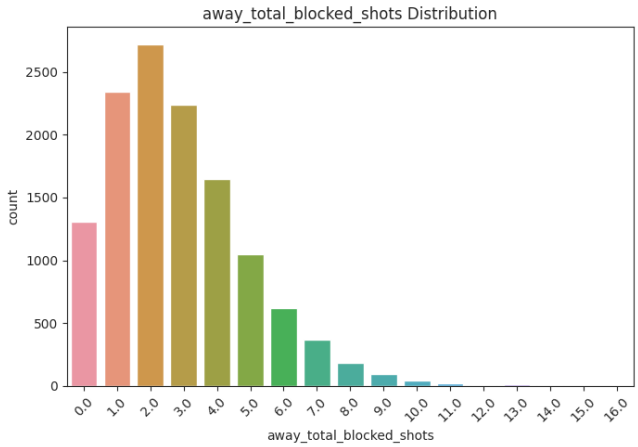






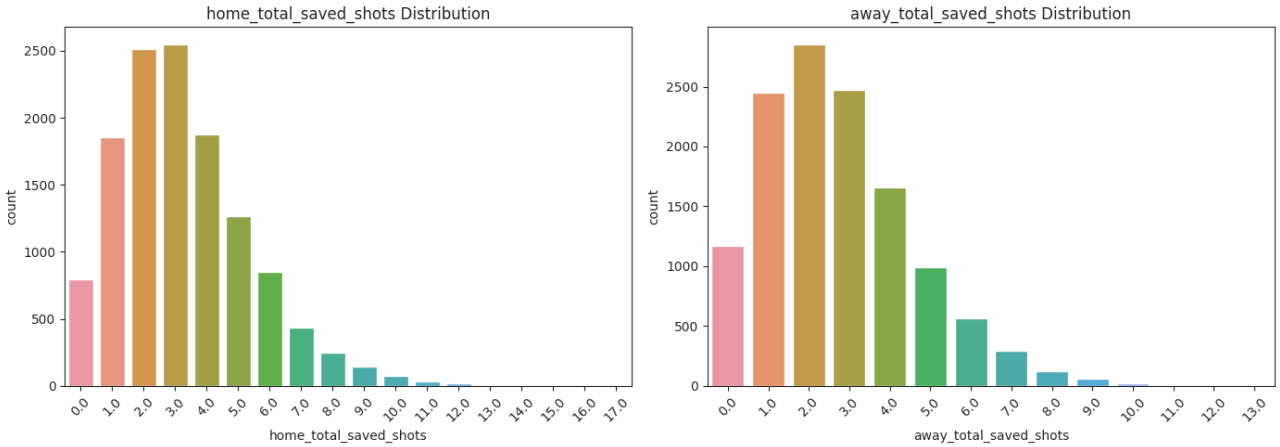


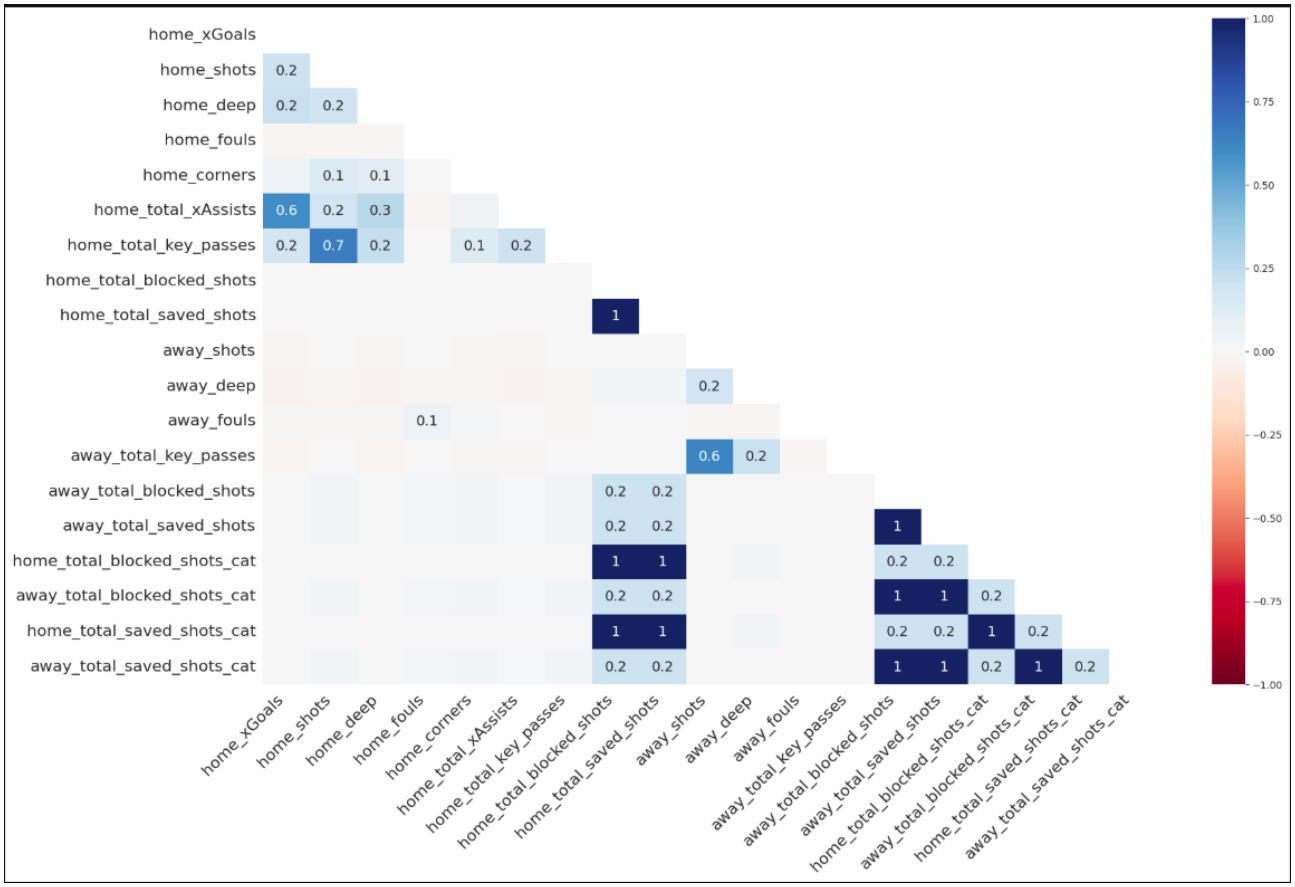




Football Match Result Prediction

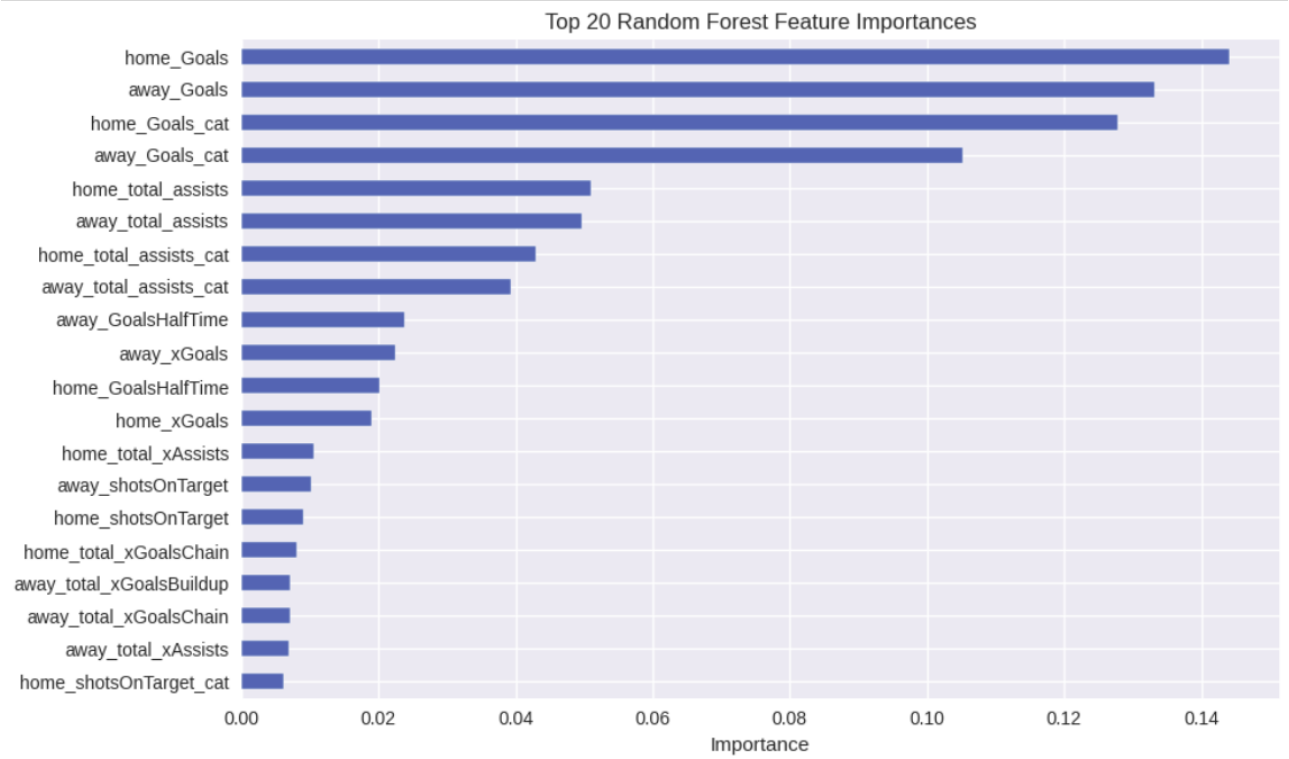
Machine Learning Project | Bar-Ilan University

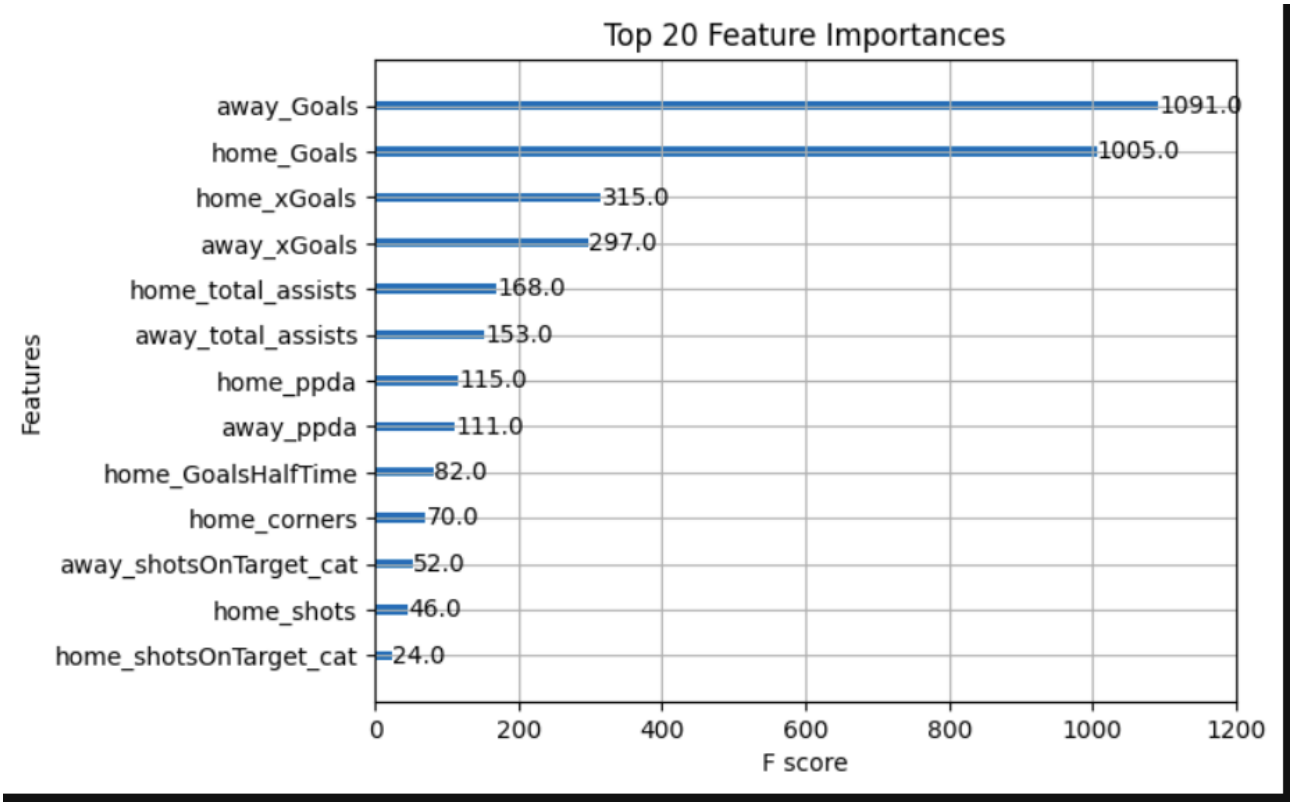




Football Match Result Prediction

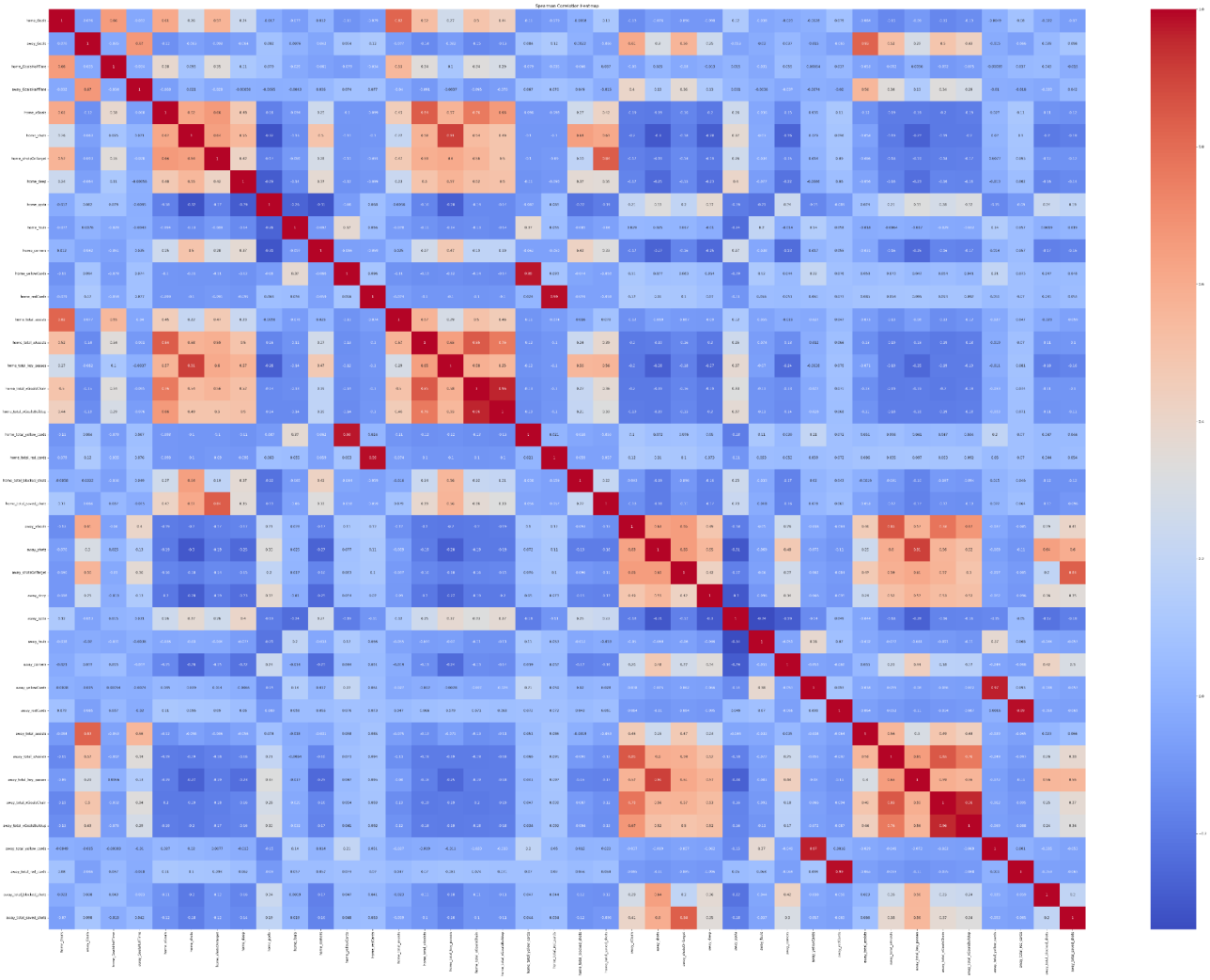
Machine Learning Project | Bar-Ilan University





Football Match Result Prediction

Machine Learning Project | Bar-Ilan University



5. Models Used

We tested various classifiers: Logistic Regression, SVM, Decision Trees, Random Forest, Gradient Boosting, AdaBoost, and XGBoost. The target variable was well-balanced, and no resampling was required.

6. Final Model Deployment

After tuning, XGBoost achieved the best results and was selected as the final model:

```
XGBClassifier(
  learning_rate=0.05,
  max_depth=110,
  min_child_weight=50,
```

```
  subsample=0.8,
  n_estimators=400,
  objective='multi:softprob',
  eval_metric='logloss',
  random_state=0
)
```

This model was stable across training, dev, and test datasets and is ready for production.

Thank you

Leonardo Romano

Notebook Stages

1. Data Preprocessing
2. EDA (AutoViz)
3. EDA (Manual)
4. Outliers and Missing Values
5. Feature Engineering
6. Feature Selection
7. Classification Model and Hyperparameter Finetuning